# Optimal parallel quantum query algorithms*

Stacey Jeffery[†1], Frederic Magniez[‡2], and Ronald de Wolf[§3]

[1]David R. Cheriton School of Computer Science and Institute for Quantum Computing, University of Waterloo, Canada
[2]CNRS, LIAFA, Univ Paris Diderot, Sorbonne Paris-Cité, 75205 Paris, France
[3]CWI and University of Amsterdam, the Netherlands

**Abstract**

We study the complexity of quantum query algorithms that make $p$ queries in parallel in each timestep. This model is motivated by the fact that decoherence times of qubits are typically small, so it makes sense to parallelize quantum algorithms as much as possible. We show tight bounds for a number of problems, specifically $\Theta((n/p)^{2/3})$ $p$-parallel queries for element distinctness and $\Theta((n/p)^{k/(k+1)})$ for $k$-sum. Our upper bounds are obtained by parallelized quantum walk algorithms, and our lower bounds are based on a relatively small modification of the adversary lower bound method, combined with recent results of Belovs et al. on learning graphs. We also prove some general bounds, in particular that quantum and classical $p$-parallel complexity are polynomially related for all total functions when $p$ is not too large.

*Partially supported by the French ANR Blanc project ANR-12-BS02-005 (RDAM), a Vidi grant from the Netherlands Organization for Scientific Research (NWO), and the European Commission IST STREP projects Quantum Computer Science (QCS) 255961, Quantum Algorithms (QALGO) 600700 and the US ARO.

[†]sjeffery@uwaterloo.ca
[‡]frederic.magniez@univ-paris-diderot.fr
[§]rdewolf@cwi.nl

# 1 Introduction

**Background.** Using quantum effects to speed up computation has been a prominent research-topic for the past two decades. Most known quantum algorithms have been developed in the setting of quantum query complexity, which is the quantum generalization of decision tree complexity. Here an algorithm is charged for each "query" to the input-elements, while intermediate computation is free (see [15] for more details). This model facilitates the proof of lower bounds, and often, though not always, quantum query upper bounds carry over to quantum time complexity. For certain specific functions one can obtain large quantum-speedups in this model. For example, Grover's algorithm [23] can search an $n$-bit database (looking for a bit-position of a 1) using $O(\sqrt{n})$ queries, while any classical algorithm needs $\Omega(n)$ queries. If one considers partial functions there are even exponential speed-ups [19, 34, 33, 7].

A more recent crop of quantum speed-ups have come from algorithms based on *quantum walks*. Such algorithms typically solve a search problem by embedding the search on a graph, with "marked" vertices, which contain a solution, and doing a quantum walk on this graph that converges rapidly to a superposition over only the marked vertices. An important example is Ambainis's quantum algorithm for solving the *element distinctness* problem [3]. In this problem one is given query access to an input $x \in [q]^n$, and the goal is to find a pair of distinct $i$ and $j$ in $[n]$ such that $x_i = x_j$, or report that none exists. Ambainis's quantum walk solves this in $O(n^{2/3})$ queries, which is optimal [1]. Classically, $\Theta(n)$ queries are required. Two generalizations of this are the *$k$-distinctness* problem, where the objective is to find distinct $i_1, \ldots, i_k \in [n]$ such that $x_{i_1} = \cdots = x_{i_k}$, and the *$k$-sum* problem, where the objective is to find distinct $i_1, \ldots, i_k \in [n]$ such that $x_{i_1} + \cdots + x_{i_k} = 0 \mod q$. Ambainis's approach solves both problems using $O(n^{k/(k+1)})$ quantum queries. Recently, Belovs gave a better $o(n^{3/4})$-query algorithm for $k$-distinctness for any fixed $k$ [8] (which can be made also time-efficient for $k = 3$ [11]). In contrast, Belovs and Špalek proved that Ambainis's $O(n^{k/(k+1)})$-query algorithm is optimal for $k$-sum [10, 14].

**Parallelization.** Here we consider to what extent such algorithms can be *parallelized*. Doing operations in parallel is a well-known way to trade hardware for time, speeding up computations by distributing the work over many processors that run in parallel. This is becoming ever more prominent in classical computing due to multi-core processors and grid computing. In the case of quantum computing there is an additional reason to consider parallelization, namely the limited lifetime of qubits due to *decoherence*: because of unintended interaction with their environment, qubits tend to lose their quantum properties over a limited amount of time, called the *decoherence time*, and degrade to classical random bits. One way to fight this is to apply the recipes of quantum error-correction and fault-tolerance[1], which can counteract the effects of sufficiently well-behaved decoherence. Another way is to try to parallelize as much as possible, so that the computation is completed before the qubits have decohered too much.

We know of only a few results about parallel quantum algorithms, most of them in the circuit model where "time" is measured by the depth of the circuit. A particularly important and beautiful example is the work of Cleve and Watrous [17], who showed how to implement the $n$-qubit quantum Fourier transform using a quantum circuit of depth $O(\log n)$. As a consequence, they were able to parallelize the quantum component of Shor's algorithm: they showed that one can factor $n$-bit integers by means of an $O(\log n)$-depth quantum circuit with polynomial-time classical pre- and post-processing. There have also been a number of papers about quantum versions of small-depth classical Boolean circuit classes like AC and NC [29, 21, 25, 35]. Beals et al. [5] show how the quantum circuit model can be efficiently simulated by

[1] It is known that parallelism is in fact *necessary* to do quantum error-correction against a constant noise rate—because noise happens in parallel, sequential operations cannot keep up with the build-up of errors.

the more realistic model of a distributed quantum computer (see also [22]). Another example, the only one we know of in the setting of query complexity, is Zalka's tight analysis of parallelizing quantum search [36, Section 4]. Suppose one wants to search an $n$-bit database, with the ability to do $p$ queries in parallel in one time-step. An easy way to make use of this parallelism is to view the database as $p$ databases of $n/p$ bits each, and to run a separate copy of Grover's algorithm on each of those. This finds a 1-position with high probability using $O(\sqrt{n/p})$ $p$-parallel queries, and Zalka showed that this is optimal up to a constant factor.

**Our results.** Here we focus on parallel quantum algorithms in the setting of quantum query complexity. Consider a function $f : \mathcal{D} \to \{0,1\}$, with $\mathcal{D} \subseteq [q]^n$. For the standard (sequential) query complexity, we let $Q(f)$ denote the bounded-error quantum query complexity of $f$ on every input $x \in \mathcal{D}$. In the $p$-parallel query model, for some integer $p \geq 1$, an algorithm can make up to $p$ quantum queries in parallel in each timestep. In that case, we let $Q^{p\|}(f)$ denote the bounded-error $p$-parallel complexity of $f$. As always in query complexity, all intermediate input-independent computation is free. Note that $Q(f)/p \leq Q^{p\|}(f) \leq Q(f)$ for every function.

For example, it is well-known that we can compute the parity of 2 bits using one quantum query [16], hence for the $n$-bit parity function we have $Q^{p\|}(f) \leq \lceil n/2p \rceil$. Since $n/2 \leq Q(f)$ for parity [6, 20], that upper bound is tight. As mentioned above, Zalka [36] showed that $Q^{p\|}(f) = \Theta(\sqrt{n/p})$ if $f$ is the $n$-bit OR function (or the corresponding search problem). An extreme case of the parallel model is where we set $p$ large enough so that $Q^{p\|}(f)$ becomes 1; such algorithms are called "nonadaptive," because they make all their queries in parallel, not adapting them to the results of earlier queries. Montanaro [28] showed that such nonadaptive quantum algorithms cannot improve much over classical algorithms: every Boolean function that depends on $n$ input bits needs $p \geq n/2$ nonadaptive quantum queries for exact computation, and $p \geq \Omega(n)$ for bounded-error computation.

In the next few sections we will prove matching upper and lower bounds on the $p$-parallel complexity $Q^{p\|}(f)$ for a number of more complicated problems: $\Theta((n/p)^{2/3})$ queries for element distinctness and $\Theta((n/p)^{k/(k+1)})$ for the $k$-sum problem.[2] Our upper bounds are obtained by parallelized quantum walk algorithms, and our lower bounds are based on a modification of the adversary lower bound method combined with some recent results by Belovs et al. about using so-called "learning graphs", both for upper and for lower bounds [9, 13, 10, 14]. The modification we need to make is surprisingly small, and technically we need to do little more than adapt the recent progress on sequential algorithms to the parallel case. Still, we feel this extension is important because (1) we are the first to do so, (2) parallel quantum algorithms are important and yet received little attention before, and (3) the fact that the extension is easy and natural increases our confidence that the adversary method combined with learning graphs is the "right" approach in the sequential as well as the parallel case.

Finally, in Section 5 we prove some more "structural" results, i.e., bounds for $Q^{p\|}(f)$ that hold for all Boolean functions $f : \{0,1\}^n \to \{0,1\}$. Specifically, based on earlier results in the sequential model due to Beals et al. [6], we show that if $p$ is not too large then $Q^{p\|}(f)$ is polynomially related to its classical deterministic $p$-parallel counterpart. We also observe that $Q^{p\|}(f) \approx n/2p$ for almost all $f$.

## 2 Preliminaries

We use $[n] := \{1, \ldots, n\}$, $\binom{[n]}{k} := \{S \subseteq [n] : |S| = k\}$, $\binom{[n]}{\leq k} := \{S \subseteq [n] : |S| \leq k\}$, and $\binom{n}{\leq k} := |\binom{[n]}{\leq k}| = \sum_{s=0}^{k} \binom{n}{s}$.

[2] The constant hidden in the $\Theta$ depends on $k$.

**Sequential and parallel query complexity.** In this paper we focus on parallel quantum algorithms in the setting of quantum query complexity. In the $p$-parallel setting, in each timestep we can make up to $p$ such queries in parallel. Precisely, a query to an input $x \in [q]^n$ corresponds to the following unitary map on two quantum registers:

$$|i, b\rangle \mapsto |i, b + x_i\rangle.$$

Here the first $n$-dimensional register contains the index $i \in [n]$ of the queried element, and the value of that element is added (in $\mathbb{Z}_q$) to the contents of the second ($q$-dimensional) register. It might be important for an algorithm to not make a query at all for a part of its superposition state. This will be even more relevant for the parallel model. In order to enable this we extend the previous unitary map to the case $i = 0$ by

$$|0, b\rangle \mapsto |0, b\rangle.$$

In each timestep we can make up to $p$ quantum queries in parallel, each on its own two registers. As always in query complexity, all intermediate input-independent computation is free.

Consider a function $f : \mathcal{D} \to \{0, 1\}$, with $\mathcal{D} \subseteq [q]^n$. When $p = 1$ we have the standard sequential query complexity, and we let $Q_\varepsilon(f)$ denote the quantum query complexity of $f$ with error probability $\leq \varepsilon$ on every input $x \in \mathcal{D}$. For general $p$, let $Q_\varepsilon^{p\|}(f)$ be the $p$-parallel complexity of $f$. Note that $Q_\varepsilon(f)/p \leq Q_\varepsilon^{p\|}(f) \leq Q_\varepsilon(f)$ for every function. The exact value of the error probability $\varepsilon$ does not matter, as long as it is a constant $< 1/2$. We usually fix $\varepsilon = 1/3$, abbreviating $Q(f) = Q_{1/3}(f)$ and $Q^{p\|}(f) = Q_{1/3}^{p\|}(f)$ as already used in the introduction.

We will use an extension of the adversary bound for the usual sequential (= 1-parallel) quantum query model. An *adversary matrix* $\Gamma$ for $f$ is a real-valued matrix whose rows are indexed by $f^{-1}(0)$ and whose columns are indexed by $f^{-1}(1)$.[3] Let $\Delta_j$ be the Boolean matrix whose rows are indexed by $x \in f^{-1}(0)$ and whose columns are indexed by $y \in f^{-1}(1)$, and such that $\Delta_j[x, y] = 1$ if $x_j \neq y_j$, and $\Delta_j[x, y] = 0$ otherwise. The (negative-weights) adversary bound for $f$ is given by the following expression:

$$\mathrm{ADV}(f) = \max_\Gamma \frac{\|\Gamma\|}{\max_{j\in[n]} \|\Gamma \circ \Delta_j\|}, \tag{1}$$

where $\Gamma$ ranges over all adversary matrices for $f$, '$\circ$' denotes entry-wise product of two matrices, and '$\|\cdot\|$' denotes the operator norm associated to the $\ell_2$ norm. This lower bound was introduced by Høyer et al. [24], generalizing Ambainis [2].[4] They showed

$$Q_\varepsilon(f) \geq \frac{1}{2}(1 - \sqrt{\varepsilon(1-\varepsilon)})\mathrm{ADV}(f). \tag{2}$$

Recently, Reichardt et al. [32, 26] showed this lower bound is actually tight: $Q(f) = \Theta(\mathrm{ADV}(f))$ for all $f$.

**Quantum walks.** We will analyze our algorithms in the quantum walk framework of [27], which we now briefly describe. Given a reversible Markov process $P$ on state space $V$, and a subset $M \subset V$ of marked elements, we define three costs: the setup cost, $\mathsf{S}$, is the cost to construct a superposition over all states $\sum_{v\in V} \sqrt{\pi_v}|v\rangle$, where $\pi$ is the stationary distribution of $P$; the checking cost, $\mathsf{C}$, is the cost to check if a

[3] One also often sees this defined as a matrix whose rows and columns are both indexed by the set all inputs, and that is required to be 0 on $x, y$-entries where $f(x) = f(y)$. Both definitions of an adversary matrix give the same lower bound.

[4] It is often denoted $\mathrm{ADV}^{\pm}(f)$ instead of $\mathrm{ADV}(f)$, but we will later use superscript to indicate parallelism, so we drop the '$\pm$' in order to prevent too many superscripts.

state $v \in V$ is in $M$; and the update cost, $\mathsf{U}$, is the cost to perform the map $|v\rangle|0\rangle \mapsto |v\rangle \sum_{u \in V} \sqrt{P_{vu}}|u\rangle$. Then, if $\delta$ is the spectral gap of $P$, and $\varepsilon$ is a lower bound on $\sum_{v \in M} \pi_v$ whenever $M$ is nonempty, we can determine if $M$ is nonempty with bounded error probability in cost

$$O\left(\mathsf{S} + \frac{1}{\sqrt{\varepsilon}}\left(\frac{1}{\sqrt{\delta}}\mathsf{U} + \mathsf{C}\right)\right).$$

If $\mathsf{S}$, $\mathsf{U}$ and $\mathsf{C}$ denote query complexities, then the above expression gives the bounded-error query complexity of the quantum walk algorithm. If, instead, these three costs denote $p$-parallel query complexities, then the above expression gives the bounded-error $p$-parallel query complexity of the quantum walk algorithm.

# 3 Lower bounds for parallel quantum query complexity

## 3.1 Adversary bound for parallel algorithms

We start by extending the adversary bound for the usual sequential quantum query algorithms to $p$-parallel algorithms. For $J \subseteq [n]$, let $x_J$ be the string $x$ restricted to the entries in $J$. Let $\Delta_J$ be the Boolean matrix whose rows are indexed by $x \in f^{-1}(0)$ and whose columns are indexed by $y \in f^{-1}(1)$, and that has a 1 at position $(x, y)$ iff $x_J \neq y_J$ (i.e., $x_j \neq y_j$ for at least one $j \in J$). For $J = \emptyset$, $\Delta_J$ is the all-0 matrix. Define the following quantity:

$$\mathrm{ADV}^{p\|}(f) = \max_{\Gamma} \frac{\|\Gamma\|}{\max_{J \in \binom{[n]}{\leq p}} \|\Gamma \circ \Delta_J\|}. \tag{3}$$

The following fact implies that we only need to consider sets $J \in \binom{[n]}{p}$ in the above definition.

**Fact 1** *For every set $J \subseteq K \subseteq [n]$, we have $\|\Gamma \circ \Delta_J\| \leq 2\|\Gamma \circ \Delta_K\|$.*

**Proof.** We use the $\gamma_2$-norm for matrices, which is defined as follows:

$$\gamma_2(A) = \min_{X,Y : A = XY} r(X)c(Y),$$

where $r(X)$ denotes the maximum squared length among the rows of $X$, and $c(Y)$ denotes the maximum squared length among the columns of $Y$. Note that the identity and the all-1 matrix both have $\gamma_2$-norm equal to 1, and $\gamma_2(A \otimes B) = \gamma_2(A)\gamma_2(B)$. Since $\Delta_J$ can be written as the all-1 matrix of the appropriate dimensions, minus identity tensored with a smaller all-1 matrix, the triangle inequality implies $\gamma_2(\Delta_J) \leq 2$. The $\gamma_2$-norm satisfies $\|A \circ B\| \leq \|A\|\gamma_2(B)$ by [26, Lemma A.1]. Observe that $\Gamma \circ \Delta_J = (\Gamma \circ \Delta_K) \circ \Delta_J$. Hence we have

$$\|\Gamma \circ \Delta_J\| = \|(\Gamma \circ \Delta_K) \circ \Delta_J\| \leq \|\Gamma \circ \Delta_K\|\gamma_2(\Delta_J) \leq 2\|\Gamma \circ \Delta_K\|.$$

□

Therefore we also have the following alternative definition, up to a multiplicative constant,

$$\mathrm{ADV}^{p\|}(f) = \max_{\Gamma} \frac{\|\Gamma\|}{\max_{J \in \binom{[n]}{p}} \|\Gamma \circ \Delta_J\|}.$$

**Theorem 2** *For every $f : \mathcal{D} \to \{0,1\}$, with $\mathcal{D} \subseteq [q]^n$, we have $Q^{p\|}(f) = \Theta(\mathrm{ADV}^{p\|}(f))$.*

**Proof.** In order to derive $p$-parallel lower bounds from sequential lower bounds, observe that we can make a bijection between input $x \in [q]^n$ and a larger string $X$ indexed by all sets $J \in \binom{[n]}{\leq p}$, such that $X_J = (x_j)_{j \in J}$. That is, each index $J$ of $X$ corresponds to up to $p$ indices $j$ of $x$. We now define a new function $F : \mathcal{D}' \to \{0,1\}$, where $\mathcal{D}'$ is the set of $X$ as above, in 1-to-1 correspondence with the elements of $x \in \mathcal{D}$, and $F(X)$ is defined as $f(x)$.[5] One query to $X$ can be simulated by $p$ parallel queries to $x$, and vice versa, so we have

$$Q^{p\|}(f) = Q(F).$$

We have $Q(F) = \Theta(\mathrm{ADV}(F))$ by [32, 26]. Now Eq. (1) applied to $F$ gives the claimed lower bound of Eq. (3) on $Q^{p\|}(f)$. □

Sometimes we can even use the same adversary matrix $\Gamma$ to obtain optimal lower bounds for $F$ as well as for $f$. A simple example of this is the $n$-bit OR-function. Let $\Gamma$ be the $n$-dimensional all-ones $1 \times n$ matrix, with the row corresponding to input $0^n$ and the columns indexed by all weight-1 inputs. Then $\|\Gamma\| = \sqrt{n}$ and $\|\Gamma \circ \Delta_j\| = 1$ for all $j \in [n]$, and hence $Q(\mathrm{OR}) = \Omega(\mathrm{ADV}(\mathrm{OR})) = \Omega(\sqrt{n})$. To get $p$-parallel lower bounds, we define a new function $F : X \mapsto \{0,1\}$ as in the proof of Theorem 2. We can use exactly the same adversary matrix $\Gamma$, with the $n$ columns still indexed by the weight-1 inputs to $f$ (which induce 1-inputs to $F$). Now $J$ will range over all subsets of $[n]$ of size at most $p$, and $\Delta_J$ will be the matrix whose $(x,y)$-entry is 1 if there is at least one $j \in J$ such that $x_j \neq y_j$. Note that $\|\Gamma \circ \Delta_J\| = \sqrt{|J|}$ for all $J$. Hence $Q^{p\|}(\mathrm{OR}) = \Omega(\mathrm{ADV}(F)) = \Omega(\sqrt{n/p})$. This is optimal and was already proved by Zalka [36, Section 4].

### 3.2 Belovs's learning graph approach

Recently Belovs [9] gave a new approach to designing quantum algorithms via the optimality of the adversary method: he introduced so-called *learning graphs* to prove upper bounds on the adversary bound, and hence upper bounds on quantum query complexity. We state it here for *certificate structures*. These are defined as follows, slightly simplified compared to Definitions 1 and 3 of Belovs and Rosmanis [13] (in particular, for us $M$ denotes a minimal certificate, while in [13] it denotes the set of supersets of a minimal certificate).

**Definition 1** *Let $\mathcal{C}$ be a set of incomparable subsets of $[n]$. We say $\mathcal{C}$ is a* 1-certificate structure *for a function $f : \mathcal{D} \to \{0,1\}$, with $\mathcal{D} \subseteq [q]^n$, if for every $x \in f^{-1}(1)$ there exists an $M \in \mathcal{C}$ such that for all $y \in \mathcal{D}$, $y_M = x_M$ implies $f(y) = 1$. We say $\mathcal{C}$ is $k$-*bounded *if $|M| \leq k$ for all $M \in \mathcal{C}$.*

The learning graph complexity of $\mathcal{C}$ is defined as follows, in its primal formulation as a minimization problem (we will see an equivalent dual formulation soon). Let $\mathcal{E} = \{(S,j) : S \subseteq [n], j \in [n] \backslash S\}$. For $e = (S,j) \in \mathcal{E}$, we use $s(e) = S$ and $t(e) = S \cup \{j\}$.

[5] Note that for $p > 1$ the new function $F$ is partial, even if the underlying $f$ is a total function.

$$\mathrm{LGC}(\mathcal{C}) = \min \sqrt{\sum_{e\in\mathcal{E}} w_e} \tag{4}$$

$$\text{s.t.} \sum_{e\in\mathcal{E}} \frac{\theta_e(M)^2}{w_e} \leq 1 \qquad \text{for all } M \in \mathcal{C} \tag{5}$$

$$\sum_{e\in\mathcal{E}:t(e)=S} \theta_e(M) = \sum_{e\in\mathcal{E}:s(e)=S} \theta_e(M) \qquad \text{for all} M \in \mathcal{C}, \emptyset \neq S \subseteq [n], M \not\subseteq S \tag{6}$$

$$\sum_{e=(\emptyset,j)\in\mathcal{E}} \theta_e(M) = 1 \qquad \text{for all } M \in \mathcal{C} \tag{7}$$

$$\theta_e(M) \in \mathbb{R}, w_e \geq 0 \qquad \text{for all } e \in \mathcal{E} \text{ and } M \in \mathcal{C} \tag{8}$$

Conditions (6) and (7) define the notions of *flow* and *unit flow*. For each $M$, $\theta_e(M)$ is a *flow* from $\emptyset$ to $M$ on the graph with vertices $\{S \subseteq [n]\}$ and edges $\{\{S, S\cup\{j\}\} : (S, j) \in \mathcal{E}\}$ if $\theta_e(M)$ satisfies condition (6). Moreover, $\theta_e(M)$ is a *unit flow* if it also satisfies condition (7).

Belovs showed that the learning graph complexity of $\mathcal{C}$ provides an upper bound on $\mathrm{ADV}(f)$, and hence on $Q(f)$, for any function $f$ having that same certificate structure. This upper bound is not always optimal, because it only uses part of the description of the function, namely its 1-certificate structure. For example the $k$-distinctness problem has quantum query complexity $o(n^{3/4})$ [8], even though it has the same 1-certificate structure as the $k$-sum problem, whose quantum query complexity is $\Theta(n^{k/(k+1)})$ [10, 14].

However, Belovs and Rosmanis [13] proved that for the special class of functions induced by $\mathcal{C}$ combined with an *orthogonal array*, it turns out the upper bound $\mathrm{LGC}(\mathcal{C})$ is optimal.

**Definition 2** *An* orthogonal array *of length $k$ is a set $T \subseteq [q]^k$, such that for every $i \in [k]$ and every $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k$ there exists exactly one $x_i \in [q]$ such that $(x_1, \ldots, x_k) \in T$.*

**Theorem 3 (Belovs-Rosmanis)** *Let $\mathcal{C}$ be a 1-certificate structure, $q \geq 2|\mathcal{C}|$, and let each $M \in \mathcal{C}$ be equipped with an orthogonal array $T_M$ of length $|M|$. Define a Boolean function $f : [q]^n \to \{0, 1\}$ as follows: $f(x) = 1$ iff there exists an $M \in \mathcal{C}$ such that $x_M \in T_M$. Then $Q(f) = \Theta(\mathrm{LGC}(\mathcal{C}))$.*

For example, the element distinctness problem ED on input $x \in [q]^n$ is defined to be 1 iff there exist distinct $i, j \in [n]$ such that $x_i = x_j$. This function is induced by the 2-bounded 1-certificate structure $\mathcal{C} = \binom{[n]}{2}$, equipped with associated orthogonal arrays $T_{\{i,j\}} = \{(v, v) : v \in [q]\}$. Hence $Q(\mathrm{ED}) = \Theta(\mathrm{LGC}(\mathcal{C}))$.

Belovs and Rosmanis [13] use duality of convex programs to show that an equivalent dual definition of the learning graph complexity as a maximization problem is the following:

$$\mathrm{LGC}(\mathcal{C}) = \max \sqrt{\sum_{M\in\mathcal{C}} \alpha_\emptyset(M)^2} \tag{9}$$

$$\text{s.t.} \sum_{M\in\mathcal{C}} (\alpha_{s(e)}(M) - \alpha_{t(e)}(M))^2 \leq 1 \qquad \text{for all } e \in \mathcal{E} \tag{10}$$

$$\alpha_S(M) = 0 \qquad \text{whenever } M \subseteq S$$

$$\alpha_S(M) \in \mathbb{R} \qquad \text{for all } S \subseteq [n] \text{ and } M \in \mathcal{C}$$

In particular, that means we can prove *lower* bounds on $\mathrm{LGC}(\mathcal{C})$ (and hence, for the functions described in Theorem 3, on $Q(f)$) by exhibiting a feasible solution $\{\alpha_S(M)\}$ for this maximization problem and calculating its objective value.

Before stating a similar result for $p$-parallel query complexity, we first adapt learning graphs. Previously, edges were of type $e = (S, j)$, where $S \subseteq [n]$ and $j \in [n] \setminus S$. Now edges are of type $e = (S, J)$, where $S \subseteq [n]$, $J \subseteq [n] \setminus S$ and $|J| \leq p$.

**Definition 3** *The* $p$-parallel learning graph complexity $\mathrm{LGC}^{p\|}(\mathcal{C})$ *of* $\mathcal{C}$ *is defined as* $\mathrm{LGC}(\mathcal{C})$ *where we replace the edge set* $\mathcal{E}$ *with* $\mathcal{E}_p = \{(S, J) : S \subseteq [n], J \subseteq [n] \setminus S, |J| \leq p\}$.

*Its dual expression is analogous. In particular, constraint* (10) *is modified to*

$$\sum_{M \in \mathcal{C}} (\alpha_{s(e)}(M) - \alpha_{t(e)}(M))^2 \leq 1 \text{ for all } e = (S, J) \in \mathcal{E}_p,$$

*where* $s(e) = S$ *and* $t(e) = S \cup J$. *We will refer to this modified constraint as "parallel-*(10)*."*

As for the special case of $p = 1$, the $p$-parallel learning graph complexity of $\mathcal{C}$ provides an upper bound on $\mathrm{ADV}^{p\|}(f)$, and hence on $Q^{p\|}(f)$, for any function $f$ having that same certificate structure.

**Lemma 4** *Let* $\mathcal{C}$ *be a certificate structure for a function* $f$. *Then* $\mathrm{ADV}^{p\|}(f) \leq \mathrm{LGC}^{p\|}(\mathcal{C})$.

**Proof.** The proof is a straightforward adaptation of the proof of [12, Theorem 9], but we repeat it here for completeness. Let $\{w_{S,J} : (S, J) \in \mathcal{E}_p\}$ and $\{\theta_{S,J}(M) : (S, J) \in \mathcal{E}_p, M \in \mathcal{C}\}$ be an optimal solution to the primal formulation of $\mathrm{LGC}^{p\|}(\mathcal{C})$.

We will use this solution to construct a feasible solution to the dual expression of our $p$-parallel adversary of Eq. (3), which is the following:

$$\begin{aligned} \mathrm{ADV}^{p\|}(f) = \min & \sqrt{\max_{x \in [q]^n} \sum_{J \in \binom{[n]}{\leq p}} \| |u_{x,J}\rangle \|^2} && (11) \\ \text{s.t. } & |u_{x,J}\rangle \in \mathbb{C}^k && \text{for all } x \in [q]^n, J \in \binom{[n]}{\leq p} \\ & \sum_{J : x_J \neq y_J} \langle u_{x,J} | u_{y,J} \rangle = 1 && \text{for all } x \in f^{-1}(1), y \in f^{-1}(0) \end{aligned}$$

The dimension $k$ of the vectors $|u_{x,J}\rangle$ can be anything, and is implicitly minimized over.

For each $x \in f^{-1}(1)$, let $M_x \in \mathcal{C}$ be such that for every $y \in [q]^n$, $x_{M_x} = y_{M_x}$ implies $f(y) = 1$. For every $x \in \mathcal{D}$ and $J \in \binom{[n]}{\leq p}$, define the following state in $\mathrm{span}\{|S\rangle|\alpha\rangle : S \subseteq [n], \alpha \in [q]^S\}$:

$$|u_{x,J}\rangle := \begin{cases} \sum_{S \subseteq [n] \setminus J} \sqrt{w_{S,J}} |S, x_S\rangle & \text{if } f(x) = 0 \\ \sum_{S \subseteq [n] \setminus J} \frac{\theta_{S,J}(M_x)}{\sqrt{w_{S,J}}} |S, x_S\rangle & \text{if } f(x) = 1 \end{cases}$$

We now verify that $\{|u_{x,J}\rangle\}_{x,J}$ is a feasible solution to the dual formulation of $\mathrm{ADV}^{p\|}(f)$:

$$\begin{aligned} \sum_{J \in \binom{[n]}{\leq p} : x_J \neq y_J} \langle u_{x,J} \mid u_{y,J} \rangle &= \sum_{J \in \binom{[n]}{\leq p} : x_J \neq y_J} \sum_{S \subseteq [n] \setminus J : x_S = y_S} \frac{\theta_{S,J}(M_x)}{\sqrt{w_{S,J}}} \sqrt{w_{S,J}} && (12) \\ &= \sum_{S \subseteq [n] : x_S = y_S} \sum_{J \in \binom{[n] \setminus S}{\leq p} : x_J \neq y_J} \theta_{S,J}(M_x). && (13) \end{aligned}$$

To see that this expression is equal to 1, we need only notice that Eq. (13) is the sum of the flow on all edges across the cut induced by the set $\{S \subseteq [n] : x_S = y_S\}$, and the total flow across a cut is always 1, since $\theta(M_x)$ is a unit flow. Thus the constraint from (11) is satisfied and $\{|u_{x,J}\rangle\}_{x,J}$ is a feasible solution.

We can now lower bound $\mathrm{ADV}^{p\|}(f)$ by the objective value of the feasible solution $\{|u_{x,J}\rangle\}_{x,J}$. First note that for any $x \in f^{-1}(1)$, by constraint (5), we have:

$$\sum_{J \in \binom{[n]}{\leq p}} \||u_{x,J}\rangle\|^2 = \sum_{J \in \binom{[n]}{\leq p}} \sum_{S \subseteq [n] \setminus J} \frac{\theta_{S,J}(M_x)^2}{w_{S,J}} \leq 1.$$

We can therefore compute the objective value as:

$$\begin{aligned} \mathrm{ADV}^{p\|}(f) &\leq \sqrt{\max_{x \in [q]^n} \sum_{J \in \binom{[n]}{\leq p}} \||u_{x,J}\rangle\|^2} \leq \sqrt{\max\left\{1, \sum_{J \in \binom{[n]}{\leq p}} \sum_{S \subseteq [n] \setminus J} w_{S,J}\right\}} \\ &\leq \sqrt{\sum_{e \in \mathcal{E}_p} w_e} = \mathrm{LGC}^{p\|}(\mathcal{C}), \end{aligned}$$

where $\sum_e w_e \geq 1$ follows from $\sum_e \theta_e(M_x) = 1$, $\sum_e \frac{\theta_e(M_x)^2}{w_e} \leq 1$, and Jensen's inequality. □

We now generalize Theorem 3 to the $p$-parallel case.

**Theorem 5** *Let $\mathcal{C}$ be a certificate structure, $q \geq 2|\mathcal{C}|$, and let each $M \in \mathcal{C}$ be equipped with an orthogonal array $T_M$ of length $|M|$. Define a Boolean function $f : [q]^n \to \{0,1\}$ as follows: $f(x) = 1$ iff there exists an $M \in \mathcal{C}$ such that $x_M \in T_M$. Then $Q^{p\|}(f) = \Theta(\mathrm{LGC}^{p\|}(\mathcal{C}))$.*

**Proof.** For the upper bound, we have $Q^{p\|}(f) = O(\mathrm{LGC}^{p\|}(\mathcal{C}))$ by Theorem 2 and Lemma 4.

For the lower bound we omit the parts that follow directly from the proof of [13, Theorem 5]. In particular, we start similarly from a feasible solution to the dual (9) and construct an adversary matrix $\Gamma$ (defined in Appendix A) such that

$$\|\Gamma\| \geq \sqrt{\frac{1}{2} \sum_{M \in \mathcal{C}} \alpha_\emptyset(M)^2}.$$

The next lemma (proved in Appendix A) generalizes a result from [13] that applied to singleton $J$.

**Lemma 6** *For every $J \subseteq [n]$, the matrix $\Gamma$ satisfies $\|\Gamma \circ \Delta_J\| \leq 2 \max_{S \subseteq [n]} \sqrt{\sum_{M \in \mathcal{C}} (\alpha_S(M) - \alpha_{S \cup J}(M))^2}$.*

When $J$ has size at most $p$, the latter maximized quantity is at most 1 because of the constraint parallel-(10) (applied to edge $(S, J') \in \mathcal{E}_p$ with $J' = J \setminus S$). Therefore

$$\mathrm{ADV}^{p\|}(f) \geq \frac{\|\Gamma\|}{\max_{J \in \binom{[n]}{p}} \|\Gamma \circ \Delta_J\|} \geq \frac{1}{2\sqrt{2}} \mathrm{LGC}^{p\|}(\mathcal{C}).$$

□

# 4 Parallel quantum query complexity of specific functions

## 4.1 Algorithms

In this section we give upper bounds for element distinctness and $k$-sum in the $p$-parallel quantum query model. We show these upper bounds by giving quantum walk algorithms.

The $p$-parallel algorithm we present for element distinctness is based on the sequential query algorithm for element distinctness of Ambainis [3]. Ambainis's algorithm uses a quantum walk on a Johnson graph, $J(n,r)$, which has vertex set $V = \{S \subseteq [n] : |S| = r\}$ and edge set $\{\{S,S'\} \subseteq V : |S \setminus S'| = 1\}$. In Ambainis's algorithm each state $S \in V$ represents a set of queried indices. The algorithm seeks a state $S$ containing $(i,x_i)$ and $(j,x_j)$ such that $i \neq j$ and $x_i = x_j$. Such a vertex $S$ is said to be *marked* in $J(n,r)$.

**Theorem 7** *The element distinctness problem on* $[q]^n$ *has* $Q^{p\|}(\text{ED}) = O((n/p)^{2/3})$.

**Proof.** We modify Ambainis's quantum walk algorithm slightly to fit into the $p$-parallel query model. Consider a walk $J(n,r/p)^p$, on $p$ copies of the Johnson graph $J(n,r/p)$. Vertices are $p$-tuples $(S_1,S_2,\ldots,S_p)$ where, for each $i \in [p]$, $S_i \subseteq [n]$ and $|S_i| = r/p$. Two vertices $(S_1,S_2,\ldots,S_p)$ and $(S'_1,S'_2,\ldots,S'_p)$ are adjacent if, for each $i \in [p]$, $|S_i \setminus S'_i| = 1$. We consider a state $(S_1,S_2,\ldots,S_p)$ *marked* if a pair of colliding elements is in $\bigcup_{i=1}^p S_i$. Since the stationary distribution is $\mu^p$, where $\mu$ is the uniform distribution over subsets of $[n]$ of size $r/p$, the probability that a state is marked is at least $\varepsilon = \Omega(r^2/n^2)$.

The setup cost, in terms of $p$-parallel queries, is only $\mathsf{S} = O(r/p)$, since we must query $r$ elements in the initial superposition over all states, but we query them $p$ at a time. Similarly, now the update requires that we query and unquery $p$ elements, but we can accomplish this in two $p$-parallel queries, so $\mathsf{U} = O(1)$. Also, $\mathsf{C} = 0$. Finally, the spectral gap $\delta$ of $p$ copies of $J(n,r/p)$ is exactly the spectral gap of one copy of $J(n,r/p)$, that is $\Omega(p/r)$.

We can now calculate the $p$-parallel query complexity of element distinctness as

$$O\left(\mathsf{S} + \frac{1}{\sqrt{\varepsilon}}\left(\frac{1}{\sqrt{\delta}}\mathsf{U} + \mathsf{C}\right)\right) = O\left(\frac{r}{p} + \frac{n}{r}\left(\sqrt{\frac{r}{p}}\right)\right) = O\left(\frac{r}{p} + \frac{n}{\sqrt{rp}}\right).$$

Setting $r$ to the optimal value of $n^{2/3}p^{1/3}$ gives an upper bound of $O((n/p)^{2/3})$. □

It is straightforward to generalize our element distinctness upper bound to $k$-sum.

**Theorem 8** *The* $k$*-sum problem on* $[q]^n$ *has* $Q^{p\|}(k\text{-sum}) = O((n/p)^{k/(k+1)})$.

**Proof.** Once again, we walk on $p$ copies of $J(n,r/p)$, but now we consider a state $(S_1,S_2,\ldots,S_p)$ marked if there are queried indices $(i_1,x_{i_1}),\ldots,(i_k,x_{i_k}) \in \bigcup_{i=1}^p S_i$ such that for all $a,b \in [k]$, $i_a \neq i_b$, and $\sum_{j=1}^k x_{i_j} = 0 \pmod q$. The proportion of marked states in a 1-instance is thus at least $\varepsilon = \Omega(r^k/n^k)$. All other parameters are as in the proof of Theorem 7. We can compute the following upper bound for $k$-sum:

$$O\left(\mathsf{S} + \frac{1}{\sqrt{\varepsilon}}\left(\frac{1}{\sqrt{\delta}}\mathsf{U} + \mathsf{C}\right)\right) = O\left(\frac{r}{p} + \frac{n^{k/2}}{r^{k/2}}\left(\sqrt{\frac{r}{p}}\right)\right) = O\left(\frac{r}{p} + \frac{n^{k/2}}{r^{(k-1)/2}\sqrt{p}}\right).$$

Setting $r$ to the optimal value of $n^{k/(k+1)}p^{1/(k+1)}$ gives an upper bound of $O((n/p)^{k/(k+1)})$. □

## 4.2 Lower bounds

We now combine ideas of Section 3.2 to prove $p$-parallel lower bounds for element distinctness and $k$-sum, matching our upper bounds of Section 4.1 if the alphabet size $q$ is sufficiently large. Our proofs are generalizations of the sequential lower bounds in [13, Section 4].

**Theorem 9** *For $q \geq 2\binom{n}{2}$, element distinctness on $[q]^n$ has $Q^{p\|}(\text{ED}) = \Omega((n/p)^{2/3})$.*

**Proof.** Recall that element distinctness is induced by the 1-certificate structure $\mathcal{C} = \binom{[n]}{2}$, equipped with associated orthogonal arrays $T_{\{i,j\}} = \{(v,v) : v \in [q]\}$. By Theorem 5, it suffices to prove the lower bound on the $p$-parallel learning graph complexity of ED. For this, it suffices to exhibit a feasible solution to the dual (9) and to lower bound its objective function. Note that the elements of $\mathcal{E}$ are now of the form $(S,J)$, where $S \subseteq [n]$ and $J \subseteq [n] \setminus S$ with $|J| \leq p$. Define

$\alpha_j = \frac{1}{2n}\max((n/p)^{2/3} - j/p, 0)$
$\alpha_S(M) = 0$ if $M \subseteq S$
$\alpha_S(M) = \alpha_{|S|}$ otherwise

To show that this is a feasible solution, the only constraint we need to verify is parallel-(10). So fix a set $S \subseteq [n]$ of some size $s$, and a set $J \subseteq [n] \setminus S$ with $|J| \leq p$. Let $L$ denote the left-hand side of parallel-(10), which is a sum over all $\binom{n}{2}$ certificates $M \in \mathcal{C}$. With respect to $e = (S,J)$, there are four kinds of $M = \{i,j\}$:

1. $i,j \in S$. Then $\alpha_{t(e)}(M) = \alpha_{s(e)}(M) = 0$, so these $M$ contribute 0 to $L$.
2. $i \in S, j \in J$. There are $s|J| \leq sp$ such $M$, and each contributes $\alpha_s^2$ to $L$, because $\alpha_{s(e)}(M) = \alpha_s$ and $\alpha_{t(e)}(M) = 0$.
3. $i,j \notin S, i,j \in J$. There are $\binom{|J|}{2} \leq \binom{p}{2}$ such $M$, and each contributes $\alpha_s^2$ to $L$.
4. $i$ and/or $j \notin S \cup J$. There are $n(n-s-|J|) \leq n^2$ such $M$, and each contributes $|\alpha_s - \alpha_{s+|J|}|^2$ to $L$.

Hence $L \leq \left(sp + \binom{p}{2}\right)\alpha_s^2 + n^2|\alpha_s - \alpha_{s+|J|}|^2 \leq p(n^{2/3}p^{1/3} + p/2)\dfrac{1}{4p^{4/3}n^{2/3}} + n^2\dfrac{1}{4n^2} \leq 1,$
where we used that $\alpha_s = 0$ if $s \geq n^{2/3}p^{1/3}$, $\alpha_s \leq \alpha_0 = \frac{1}{2p^{2/3}n^{1/3}}$, and $|\alpha_s - \alpha_{s+|J|}|^2 \leq 1/4n^2$. This proves constraint parallel-(10) holds. The objective value for this feasible solution is $\sqrt{\binom{n}{2}\alpha_0^2} = \Omega((n/p)^{2/3})$. □

**Theorem 10** *For $q \geq 2\binom{n}{k}$, the $k$-sum-problem on $[q]^n$ has $Q^{p\|}(k\text{-sum}) = \Omega\left((n/p)^{k/(k+1)}\right)$.*

**Proof.** The proof strategy is the same as in Theorem 9. We now use certificate structure $\mathcal{C} = \binom{[n]}{k}$ with the orthogonal array $T = \{(v_1,\ldots,v_k) : \sum_{i=1}^k v_i = 0 \mod q\}$. This induces the $k$-sum problem in the way mentioned in Theorem 5. We define the following solution to the dual for $\text{LGC}^{p\|}(\mathcal{C})$:

$\alpha_j = \frac{1}{2n^{k/2}}\max((n/p)^{k/(k+1)} - j/p, 0)$
$\alpha_S(M) = 0$ if $M \subseteq S$
$\alpha_S(M) = \alpha_{|S|}$ otherwise

Fix some $e = (S, J)$ with $S \subseteq [n]$ of size $s$, and disjoint $J \subseteq [n]$ of size at most $p$. Again let $L$ denote the left-hand side of constraint parallel-(10). In order to establish that the above solution is feasible, we want to show $L \leq 1$. With respect to $e$, we can distinguish different kinds of $M = \{i_1, \ldots, i_k\}$, depending on $i := |M \cap S|$ and $j := |M \cap J|$:

1. $i + j < k$. There are $\binom{s}{i}\binom{|J|}{j}$ such $M$, and each contributes $\leq |\alpha_s - \alpha_{s+|J|}|^2 \leq 1/4n^k$ to $L$.

2. $i + j = k$. There are $\binom{s}{i}\binom{|J|}{j}$ such $M$, and each contributes $\alpha_s^2$ to $L$ if $i < k$, and 0 if $i = k$.

Note that $\alpha_s = 0$ if $s \geq n^{k/(k+1)}p^{1/(k+1)}$, and $\alpha_s \leq \alpha_0 = \frac{(n/p)^{k/(k+1)}}{2n^{k/2}}$. Hence we can upper bound $L$ by

$$\sum_{i=0}^{k-1}\sum_{j=0}^{k-1-i}\binom{s}{i}\binom{|J|}{j}|\alpha_s - \alpha_{s+|J|}|^2 + \sum_{i=0}^{k-1}\binom{s}{i}\binom{|J|}{k-i}\alpha_s^2$$
$$= \sum_{\ell=0}^{k-1}\binom{s+|J|}{\ell}|\alpha_s - \alpha_{s+|J|}|^2 + \binom{s+|J|}{k-1}\alpha_s^2 \leq \frac{n^{k-1}}{4n^k} + \frac{\left(n^{k/(k+1)}p^{1/(k+1)} + p\right)^{k-1}(n/p)^{2k/(k+1)}}{4n^k} \leq 1.$$

This shows that our solution is feasible. Its objective value is $\sqrt{\binom{n}{k}\alpha_0^2} = \Omega\left((n/p)^{k/(k+1)}\right)$. □

# 5 Some general bounds

In this section we will relate quantum and classical $p$-parallel complexity. For the sequential model ($p = 1$) it is known that quantum bounded-error query complexity is at best a 6th power less than classical deterministic complexity, for all total Boolean functions [6]. Here we will see to what extent we can prove a similar result for the $p$-parallel model.

We start with a few definitions, referring to [15] for more details and background. Let $f : \{0,1\}^n \to \{0,1\}$ be a total Boolean function. For $b \in \{0,1\}$, a *$b$-certificate* for $f$ is an assignment $C : S \to \{0,1\}$ to a subset $S$ of the $n$ variables, such that $f(x) = b$ whenever $x$ is consistent with $C$. The *size* of $C$ is $|S|$. The *certificate complexity $C_x(f)$ of $f$ on $x$* is the size of a smallest $f(x)$-certificate that is consistent with $x$. The *certificate complexity* of $f$ is $C(f) = \max_x C_x(f)$. The *1-certificate complexity* of $f$ is $C^{(1)}(f) = \max_{\{x|f(x)=1\}} C_x(f)$. Given an input $x \in \{0,1\}^n$ and subset $B \subseteq [n]$ of indices of variables, let $x^B$ denote the $n$-bit input obtained from $x$ by flipping all bits $x_i$ whose index $i$ is in $B$. The *block sensitivity* $bs(f, x)$ of $f$ at input $x$, is the maximal integer $k$ such that there exist disjoint sets $B_1, \ldots, B_k$ satisfying $f(x) \neq f(x^{B_i})$ for all $i \in [k]$. The *block sensitivity* of $f$ is $bs(f) = \max_x bs(f, x)$. Nisan [30] proved that

$$bs(f) \leq C(f) \leq bs(f)^2. \tag{14}$$

Via a standard reduction [31], Zalka's $\Theta(\sqrt{n/p})$ bound for the OR-function implies:

**Theorem 11** *For every $f : \{0,1\}^n \to \{0,1\}$ we have $Q^{p\|}(f) = \Omega(\sqrt{bs(f)/p})$.*

We now prove a general upper bound on deterministic $p$-parallel complexity:

**Theorem 12** *For every $f : \{0,1\}^n \to \{0,1\}$ and $p \leq C^{(1)}(f)$ we have $D^{p\|}(f) \leq \lceil C^{(1)}(f)/p \rceil bs(f)$.*

**Proof.** Beals et al. [6, Lemma 5.3] give a deterministic decision tree for $f$ that runs for at most $bs(f)$ rounds, and in each round queries all variables of a 1-certificate for the function and substitutes their values into the function. They show that this reduces the function to a constant. By parallelizing the querying of the certificate we can implement every round using at most $\lceil C^{(1)}(f)/p \rceil$ $p$-parallel steps. □

Quantum and classical $p$-parallel complexity are polynomially related if $p$ is not too big:

**Theorem 13** *For every $f : \{0,1\}^n \to \{0,1\}$, $c > 1$, $p \leq bs(f)^{1/c}$, we have $D^{p\|}(f) \leq O(Q^{p\|}(f)^{6+4/(c-1)})$.*

**Proof.** We can assume $C(f) = C^{(1)}(f)$ (else consider $1-f$). By Eq. (14) we have $p \leq bs(f)^{1/c} \leq C^{(1)}(f)$, so we can apply Theorem 12. We also have $C^{(1)}(f) \leq bs(f)^2$. Note that the assumption on $p$ is equivalent to $p \leq (bs(f)/p)^{1/(c-1)}$. Also using Theorem 11, we obtain

$$D^{p\|}(f) \leq \lceil C^{(1)}(f)/p \rceil bs(f) \leq O(bs(f)^3/p) \leq O((bs(f)/p)^{3+2/(c-1)}) \leq O(Q^{p\|}(f)^{6+4/(c-1)}).$$

□

For example, if $p \leq bs(f)^{1/3}$ then $Q^{p\|}(f)$ can be at most an 8th power smaller than $D^{p\|}(f)$. This theorem leaves open the possibility of superpolynomial gaps between $D^{p\|}(f)$ and $Q^{p\|}(f)$ for large $p$; while we do not believe this will occur for total functions, we do not know how to prove this.

We end with an observation about random Boolean functions. Van Dam [18] showed that an $n$-bit input string $x$ can be recovered with high probability using $n/2 + O(\sqrt{n})$ quantum queries. This implies $Q(f) \leq n/2 + O(\sqrt{n})$ for all $f : \{0,1\}^n \to \{0,1\}$. Ambainis et al. [4] proved an essentially matching lower bound for random functions: almost all $f$ have $Q(f) \geq (1/2 - o(1))n$. Since trivially $Q(f) \leq pQ^{p\|}(f)$, we obtain the $p$-parallel lower bound $Q^{p\|}(f) \geq (1/2 - o(1))n/p$ for almost all $f$. This result is essentially optimal, because we can straightforwardly parallelize van Dam's algorithm to compute $x$ using roughly $n/2p$ $p$-parallel quantum queries, as follows:

1. With $T = n/2 + O(\sqrt{n \log(1/\varepsilon)})$ and $B = \sum_{i=0}^{T} \binom{n}{i}$ being the number of $y \in \{0,1\}^n$ with weight $|y| \leq T$, set up the $n$-qubit superposition $\frac{1}{\sqrt{B}} \sum_{y \in \{0,1\}^n : |y| \leq T} |y\rangle$.

2. Apply the unitary $|y\rangle \mapsto (-1)^{x \cdot y} |y\rangle$. We can implement this using $\lceil T/p \rceil$ $p$-parallel queries for $|y| \leq T$: the first batch of $p$ queries would query the first $p$ positions where $y$ has a one and put the answer in the phase; the second batch would query the next $p$ positions, etc.

3. Apply a Hadamard transform to all qubits and measure.

To see the correctness of this algorithm, note that the fraction of $n$-bit strings $y$ that have weight $> T$ is $\ll \varepsilon$. Hence the state obtained in step 2 is very close to the state $\frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle$, whose Hadamard transform is exactly $|x\rangle$.

Accordingly, for this type of "quantum oracle interrogation," parallelization gives the optimal factor-$p$ speed-up. And for $p = n/2 + O(\sqrt{n})$, one $p$-parallel query suffices.

**Corollary 14** *For all $p \leq n$, almost all $f : \{0,1\}^n \to \{0,1\}$ satisfy $Q^{p\|}(f) = (1/2 \pm o(1))n/p$.*

## 6 Conclusion and future work

This paper is the first to systematically study the power and limitations of parallelism for quantum query algorithms. It is motivated in particular by the need to reduce overall computing time when running quantum algorithms on hardware with quickly decohering quantum bits.

We leave open many interesting questions for future work, for example:

- There are many other computational problems whose $p$-parallel complexity is unknown, for example finding a triangle in a graph or deciding whether two given matrices multiply to a third one. For both of these problems, however, even the sequential quantum query complexity is still open.
- We suspect Theorem 13 is not optimal, and conjecture that $D^{p\|}(f)$ and $Q^{p\|}(f)$ are polynomially related also for large $p$. Montanaro's result [28] about the weakness of maximally parallel (=nonadaptive) quantum algorithms is evidence for this. Even for the sequential model ($p = 1$) the correct bound is open; the best relation known is a 6th power [6] but the correct answer may well be a square.

**Acknowledgement.** We thank Jérémie Roland for helpful discussions.

## References

[1] S. Aaronson and Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. *Journal of the ACM*, 51(4):595–605, 2004.

[2] A. Ambainis. Quantum lower bounds by quantum arguments. *Journal of Computer and System Sciences*, 64(4):750–767, 2002.

[3] A. Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007. Earlier version in FOCS'04. quant-ph/0311001.

[4] A. Ambainis, A. Bačkurs, J. Smotrovs, and R. de Wolf. Optimal quantum query bounds for almost all Boolean functions. In *Proceedings of 30th STACS*, pages 446–453, 2013.

[5] R. Beals, S. Brierley, O. Gray, A. Harrow, S. Kutin, N. Linden, D. Shepherd, and M. Stather. Efficient distributed quantum computing. *Proceedings of the Royal Society*, A469(2153), 2013.

[6] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, 2001.

[7] N. de Beaudrap, R. Cleve, and J. Watrous. Sharp quantum vs. classical query complexity separations. *Algorithmica*, 34(4):449–461, 2002.

[8] A. Belovs. Learning-graph-based quantum algorithm for k-distinctness. In *Proceedings of 53rd IEEE FOCS*, pages 207–216, 2012.

[9] A. Belovs. Span programs for functions with constant-sized 1-certificates. In *Proceedings of 43rd ACM STOC*, pages 77–84, 2012.

[10] A. Belovs. Adversary lower bound for element distinctness, 23 Apr 2012. arXiv:1204.5074.

[11] A. Belovs, A. Childs, S. Jeffery, R. Kothari, and F. Magniez. Time-efficient quantum walks for 3-distinctness. In *Proceedings of 40th ICALP(1)*, pages 105–122, 2013.

[12] A. Belovs and T. Lee. Quantum algorithm for k-distinctness with prior knowledge on the input. Technical Report arXiv:1108.3022, arXiv, 2011.

[13] A. Belovs and A. Rosmanis. On the power of non-adaptive learning graphs. In *Proceedings of 28th IEEE CCC*, 2013. References are to arXiv:1210.3279v2.

[14] A. Belovs and R. Špalek. Adversary lower bound for the k-sum problem. In *Proceedings of 4th ITCS*, pages 323–328, 2013.

[15] H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: A survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[16] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca. Quantum algorithms revisited. In *Proceedings of the Royal Society of London*, volume A454, pages 339–354, 1998.

[17] R. Cleve and J. Watrous. Fast parallel circuits for the quantum Fourier transform. In *Proceedings of 41st IEEE FOCS*, pages 526–536, 2000.

[18] W. van Dam. Quantum oracle interrogation: Getting all information for almost half the price. In *Proceedings of 39th IEEE FOCS*, pages 362–367, 1998.

[19] D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. In *Proceedings of the Royal Society of London*, volume A439, pages 553–558, 1992.

[20] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. A limit on the speed of quantum computation in determining parity. *Physical Review Letters*, 81:5442–5444, 1998.

[21] F. Green, S. Homer, C. Moore, and C. Pollett. Counting, fanout and the complexity of quantum ACC. *Quantum Information and Computation*, 2(1):35–65, 2002.

[22] L. Grover and T. Rudolph. How significant are the known collision and element distinctness quantum algorithms? *Quantum Information and Computation*, 4(3):201–206, 2004.

[23] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of 28th ACM STOC*, pages 212–219, 1996.

[24] P. Høyer, T. Lee, and R. Špalek. Negative weights make adversaries stronger. In *Proceedings of 39th ACM STOC*, pages 526–535, 2007.

[25] P. Høyer and R. Špalek. Quantum fan-out is powerful. *Theory of Computing*, 1(1):81–103, 2005.

[26] T. Lee, R. Mittal, B. Reichardt, R. Špalek, and M. Szegedy. Quantum query complexity of state conversion. In *Proceedings of 52nd IEEE FOCS*, pages 344–353, 2011. References are to arXiv:1011.3020v2.

[27] F. Magniez, A. Nayak, J. Roland, and M. Santha. Search via quantum walk. *SIAM Journal on Computing*, 40(1):142–164, 2011.

[28] A. Montanaro. Nonadaptive quantum query complexity. *Information Processing Letters*, 110(24):1110–1113, 2010.

[29] C. Moore and M. Nilsson. Parallel quantum computation and quantum codes. *SIAM Journal on Computing*, 31(3):799–815, 2002.

[30] N. Nisan. CREW PRAMs and decision trees. *SIAM Journal on Computing*, 20(6):999–1007, 1991.

[31] N. Nisan and M. Szegedy. On the degree of Boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994.

[32] B. Reichardt. Span programs and quantum query complexity: The general adversary bound is nearly tight for every Boolean function. In *Proceedings of 50th IEEE FOCS*, pages 544–551, 2009.

[33] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[34] D. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[35] Y. Takahashi and S. Tani. Collapse of the hierarchy of constant-depth exact quantum circuits. In *Proceedings of 28th IEEE CCC*, 2013.

[36] Ch. Zalka. Grover's quantum searching algorithm is optimal. *Physical Review A*, 60:2746–2751, 1999.

# A Proof of Lemma 6

We need to go further into the details of the proof of [13, Theorem 5]. First we use a variation of the adversary bound from [14] that allows the duplication of row and column indices. Concretely, rows and columns of $\Gamma$ are now indexed by $(x,a)$ and $(y,a)$, respectively, where $x \in f^{-1}(0)$, $y \in f^{-1}(1)$ and $a$ belongs to some finite set. Then $\Delta_j$ is now defined such that $\Delta_j[(x,a),(y,b)] = 1$ if $x_j \neq y_j$, and $\Delta_j[(x,a),(y,b)] = 0$ otherwise.

Second, $\Gamma$ is the submatrix of a larger matrix $\widetilde{\Gamma}$ (defined below) that is indexed by the elements of $[q]^n \times \mathcal{C}$ and of $[q]^n$. Then $\Delta_j$ is naturally extended to all $x, y \in [q]^n$ and $M \in \mathcal{C}$ by $\Delta_j[(x,M),(y,M)] = 1$ if $x_j \neq y_j$, and $\Delta_j[(x,M),(y,M')] = 0$ otherwise. Since $\|\Gamma \circ \Delta_J\| \leq \|\widetilde{\Gamma} \circ \Delta_J\|$, we only need to upper bound the latter.

Consider the Hilbert space $\mathbb{C}^q$. Let $E_0$ denote the orthogonal projector onto the vector $\frac{1}{\sqrt{q}}(1,1,\ldots,1)$, and $E_1 = \mathrm{I} - E_0$ its orthogonal complement. For every $S \subseteq [n]$, let $E_S = \otimes_{j\in[n]} E_{s_j}$, where $s_j = 1$ if $j \in S$, and $s_j = 0$ otherwise. Note that $E_S E_{S'} = E_S$ if $S = S'$, and $E_S E_{S'} = 0$ otherwise. Define $\widetilde{\Gamma}$ as

$$\widetilde{\Gamma} = (G_M)_{M\in\mathcal{C}}, \quad \text{with } G_M = \sum_{S\subseteq[n]} \alpha_S(M) E_S,$$

where the $\alpha_S(M)$ come from a feasible solution to the dual (9). [13, Lemma 17] shows that the submatrix $\Gamma$ satisfies

$$\|\Gamma\| \geq \sqrt{\frac{1}{2}\sum_{M\in\mathcal{C}} \alpha_\emptyset(M)^2}.$$

However, upper bounding $\|\widetilde{\Gamma} \circ \Delta_J\|$ requires some additional steps. We first review the approach of [13] for the special case of $J = \{j\}$. Define a linear map $\varphi_j$ on matrix $\widetilde{\Gamma}$ by its action on blocks $E_S$, for every

$S \subseteq [n]$. First, let $\varphi$ be such that $\varphi(E_0) = E_0$ and $\varphi(E_1) = -E_0$. Then $\varphi_j(E_S) = E_{s_1} \otimes \ldots \otimes E_{s_{j-1}} \otimes \varphi(E_{s_j}) \otimes E_{s_{j+1}} \otimes \ldots \otimes E_{s_n}$. An alternative definition is

$$\varphi_j(E_S) = \begin{cases} E_S, & \text{if } j \notin S; \\ -E_{S\setminus\{j\}} & \text{otherwise.} \end{cases}$$

The map $\varphi_j$ was introduced because it satisfies $E_S \circ \Delta_j = \varphi_j(E_S) \circ \Delta_j$. This comes from the observation that $\varphi(E_1) \circ \Delta_1 = E_1 \circ \Delta_1$, since $E_1 = \mathrm{I} - E_0$ and $\mathrm{I} \circ \Delta_1 = 0$. The approach of [13] then consists of applying $\varphi_j$ to $\widetilde{\Gamma}$ before computing the norm of $\widetilde{\Gamma} \circ \Delta_j$.

We now generalize $\varphi_j$ to subsets $J \subseteq [n]$ as

$$\varphi_J(E_S) = \begin{cases} E_S, & \text{if } J \not\subseteq S; \\ -\displaystyle\sum_{S' : S\setminus J \subseteq S' \subsetneq S} E_{S'}, & \text{otherwise.} \end{cases}$$

Then $\varphi_j$ satisfies the following fact, which is an extension of the case $J = \{j\}$.

**Fact 15** *Let $J \subseteq [n]$ be any subset. Then $\widetilde{\Gamma} \circ \Delta_J = \varphi_J(\widetilde{\Gamma}) \circ \Delta_J$.*

Therefore we can upper bound $\|\widetilde{\Gamma} \circ \Delta_J\|$ by $2\|\varphi_J(\widetilde{\Gamma})\|$ using also Fact 1. It remains to compute the latter norm. We first compute $\varphi_J(G_M)$:

$$\varphi_J(G_M) = \sum_{S \subseteq [n]} \beta_S(M) E_S, \quad \text{where } \beta_S(M) = \alpha_S(M) - \alpha_{S\cup J}(M).$$

Observe that $\beta_S(M) = 0$ if $J \subseteq S$. Now rewrite $(\varphi_J(\widetilde{\Gamma}))^* \varphi_J(\widetilde{\Gamma})$ as

$$(\varphi_J(\widetilde{\Gamma}))^* \varphi_J(\widetilde{\Gamma}) = \sum_{M \in \mathcal{C}} (\varphi_J(G_M))^* \varphi_J(G_M) = \sum_{S \subseteq [n]} \left( \sum_{M \in \mathcal{C}} \beta_S(M)^2 \right) E_S.$$

Since the different $E_S$ project onto orthogonal subspaces, we can conclude

$$\|\varphi_J(\widetilde{\Gamma})\| = \sqrt{\|(\varphi_J(\widetilde{\Gamma}))^* \varphi_J(\widetilde{\Gamma})\|} = \max_{S \subseteq [n]} \sqrt{\sum_{M \in \mathcal{C}} \beta_S(M)^2}.$$